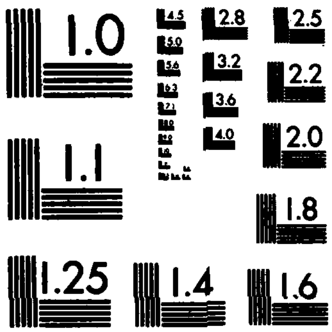
AD-A136 188 FIXED POINT IMPLEMENTATIONS OF FAST KALMAN ALGORITHMS (U) RHODE ISLAND UNIV KINGSTON DEPT OF ELECTRICAL ENGINEERING L L SCHARF ET AL. NOV 83 N00014-82-K-0300 1/1

UNCLASSIFIED F/G 12/1 NL

END
FILMED
1-84
DTIC



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

AD-A136188

(12)

FIXED POINT IMPLEMENTATIONS OF FAST KALMAN ALGORITHMS

Louis L. Scharf
Department of Electrical Engineering
University of Rhode Island

Sigurdur Sigurdsson
Department of Electrical Engineering
Colorado State University

November 1983

Prepared for

OFFICE OF NAVAL RESEARCH (Code 411 SP)
Statistics and Probability Branch
Arlington, Virginia 22217
under Contract N00014-82-K-0300
Louis L. Scharf, Principal Investigator

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER	2. GOVT ACCESSION NO. AD-A136188	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) Fixed Point Implementations of Fast Kalman Algorithms		5. TYPE OF REPORT & PERIOD COVERED Technical
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Louis L. Scharf, Sigurdur Sigurdsson		8. CONTRACT OR GRANT NUMBER(s) N00014-82-K-0300
9. PERFORMING ORGANIZATION NAME AND ADDRESS Department of Electrical Engineering, University of Rhode Island, Kingston, Rhode Island 02881		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS Office of Naval Research (Code 411 SP), Department of the Navy, Arlington, VA 22217		12. REPORT DATE ~~October~~ Nov 1983
		13. NUMBER OF PAGES
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)		15. SECURITY CLASS. (of this report)
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

DISTRIBUTION STATEMENT A
Approved for public release; Distribution Unlimited

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

In this paper we study scaling rules and round-off noise variances in a fixed point implementation of the Kalman predictor for an ARMA time series observed noise-free. The Kalman predictor is realized in a fast form that uses the so-called fast Kalman gain algorithm. The algorithm for the gain is fixed point.

Scaling rules and expressions for rounding error variances are derived. The numerical results show that the fixed point realization performs very

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

S/N 0102-LF-014-6601

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

closely to the floating point realization for relatively low-order ARMA time series that are not too narrowband.

The predictor has been implemented in 16-bit fixed point arithmetic on an INTEL 8086 microprocessor, and in 16-bit floating point arithmetic on an INTEL 8080. Fixed point code was written in ASSEMBLY language and floating point code was written in FORTRAN. Experimental results were obtained by running the fixed and floating point filters on identical data sets. All experiments were carried out on an INTEL MDS 230 Development System.

– a –

FIXED POINT IMPLEMENTATIONS OF FAST KALMAN ALGORITHMS

Louis L. Scharf — Department of Electrical Engineering, University of Rhode Island, Kingston, RI 02881

Sigurdur Sigurdsson — Department of Electrical Engineering, Colorado State University, Ft. Collins, CO 80523

4

Summary

In this paper ~~we~~ the authors study scaling rules and round-off noise variances in a fixed point implementation of the Kalman predictor for an ARMA time series observed noise-free. The Kalman predictor is realized in a fast form that uses the so-called fast Kalman gain algorithm. The algorithm for the gain is fixed point.

Scaling rules and expressions for rounding error variances are derived. The numerical results show that the fixed point realization performs very closely to the floating point realization for relatively low-order ARMA time series that are not too narrowband.

The predictor has been implemented in 16-bit fixed point arithmetic on an INTEL 8086 microprocessor, and in 16-bit floating point arithmetic on an INTEL 8080. Fixed point code was written in ASSEMBLY language and floating point code was written in FORTRAN. Experimental results were obtained by running the fixed and floating point filters on identical data sets. All experiments were carried out on an INTEL MDS 230 Development System.

Introduction

Finite-dimensional Gaussian time series have stationary Markovian state-space descriptions. In such descriptions the initial conditions are multivariate normal and state variables are predicted values of the series based on an infinite past of observations. The linear filtering problem is one of estimating the state at time t based on observations up to time t and the prediction problem is one of predicting the state at time t+1 based on observations up to time t.

Corresponding to the Markovian representation is the innovations representation. The essential characteristic of this nonstationary representation is that it may be used to synthesize a time-series, starting from zero initial conditions, whose second-order statistics match the statistics of the original time series. The states are predicted values of the time series based on a finite past of observations. Using this representation, the Kalman predictor may be written down from inspection as the causal and stable inverse of the representation. The so-called Kalman gain in the innovations representation (or equivalently in the Kalman predictor) may be associated either with the Levinson recursions for factoring the inverse of the correlation matrix for the time series or with the LeRoux-Gueguen recursions for factoring the correlation matrix itself. The latter association leads to a fixed point algorithm for computing Kalman gains. This so-called fast algorithm produces a fast Kalman filter.

This paper was presented at the 22nd IEEE Conference on Decision and Control, San Antonio, TX (December 14-16, 1983). The work was supported by Office of Naval Research, Contract # N00014-82-K-0300.

In this paper we present results from a study of fast Kalman predictors, implemented in floating point and in fixed point arithmetic, for autoregressive moving average time series. More extensive results of this study, for noisy and noise-free filtering and prediction, may be found in the thesis of Sigurdsson [1].

In our summary of results for Kalman filtering we draw heavily upon the work of Morf, Kailath, Anderson, and Moore. See [2] and [3] for our previous references to the appropriate literature. In our derivation of scaling rules and expressions for rounding error variances we adapt the stationary results of Jackson [4] and Mullis and Roberts [5] to our nonstationary problem.

Signal Models for Stationary ARMA Time Series

A zero-mean, second-order stationary, time series $\{y(t)\}$ is said to be autoregressive moving average (ARMA) if the entries $y(t)$ in the time series obey this recursion for all t:

$$\sum_{n=0}^{p} a(n)y(t-n) = \sum_{n=0}^{q} b(n)u(t-n)$$

$$E\,u(t)=0 \qquad E\,u(t)u(t+n) = \sigma^2\delta(n)$$

$$a_0 = b_0 = 1 \qquad \delta(n):\ \text{Kronecker delta}$$

Markovian Representation

The Markovian representation for $\{y(t)\}$ is summarized in the following equations, where $n = \max(p,q)$, and where for purposes of illustration we have assumed $q > p$.

State Equations

$$\underline{x}(t+1/t) = A\underline{x}(t/t-1) + \underline{h}(1)u(t)$$

$$\underline{x}(0/-1) : N(\underline{0},Q) \; ; \; u(t) = N(0,\sigma^2)$$

$$y(t) = \underline{c}'\,\underline{x}(t/t-1) + u(t)$$

Unit Pulse Response

$$h(t) = \begin{cases} 0, & t<0 \\ 1, & t=0 \\ \underline{c}'A^{t-1}\underline{h}(1), & t>0 \end{cases}$$

State Covariance

$$R(t) = A^tQ$$
$$Q = AQA' + \sigma^2\underline{h}(1)\underline{h}'(1)$$

Output Covariance

$$r(t) = \underline{c}'R(t)\,\underline{c} + \sigma^2h(t)$$

Design Equations

$$A = \begin{bmatrix} 0 & 1 & & & \\ 0 & 0 & 1 & & \\ & & & & \\ & & & 1 & \\ -a(n) & & & & -a(1) \end{bmatrix} \quad \underline{c} = \begin{bmatrix} 1 \\ 0 \\ \cdot \\ \cdot \\ 0 \end{bmatrix} \quad \underline{h}(1) = \begin{bmatrix} h(1) \\ h(2) \\ \\ \\ h(n) \end{bmatrix}$$

$$\begin{bmatrix} 1 & & & \\ a(1) & 1 & & 0 \\ \cdot & & & \\ \cdot & & & \\ \cdot & & & \\ a(p-1) & \ldots & 1 & \\ 0 & a(p-1) & & 1 \end{bmatrix} \begin{bmatrix} h(1) \\ h(2) \\ \cdot \\ \cdot \\ \cdot \\ \\ h(n) \end{bmatrix} = \begin{bmatrix} a(1) \\ a(2) \\ \cdot \\ \cdot \\ \cdot \\ a(p) \\ 0 \end{bmatrix} + \begin{bmatrix} b(1) \\ b(2) \\ \\ \\ \\ \\ b(q) \end{bmatrix}$$

Innovations Representation

The idea behind the innovations representation for $\{y(t)\}$ is to replace the stationary initial conditions, distributed as $x(0/-1) : N(\underline{0}, Q)$ with the nonstationary initial conditions $x(0/-1) = \underline{0}$, to replace the stationary input vector $\underline{h}(1)$ by a nonstationary Kalman gain vector $\underline{k}(t)$, and to replace the stationary i.i.d. sequence $\{u(t)\}$, distributed as $u(t) : N(0, \sigma^2)$, with the nonstationary i.i.d. innovations sequence $\{u(t)\}$, distributed as $u(t) : N(0, v(t))$. The trick is to choose the Kalman gain $\underline{k}(t)$ and the innovations variance $v(t)$ correctly. The innovations representation is summarized below. It is worth noting that $\underline{k}(t) \rightarrow \underline{h}(1)$, $v(t) \rightarrow \sigma^2$, and $Q(t) \rightarrow Q$ as $t \rightarrow \infty$.

State Equations

$$\underline{x}(t+1/t) = A\underline{x}(t/t-1) + \underline{k}(t)\,u(t)$$
$$\underline{x}(0/-1) = \underline{0} \; ; \; u(t) : N(0,v(t))$$
$$y(t) = \underline{c}'\,\underline{x}(t/t-1) + u(t)$$

Unit Pulse Response

$$h^t(n) = \begin{cases} 0, & n<0 \\ 1, & n=0 \\ \underline{c}'A^{n-1}\,\underline{k}(t), & n>0 \end{cases}$$

State Covariance

$$R^t(n) = A^nQ(t)$$
$$Q(t+1) = AQ(t)A' + v(t)\underline{k}(t)\underline{k}'(t)$$
$$Q(0) = 0$$

Output Covariance

$$r^t(n) = \underline{c}'R^t(n)\underline{c} + v(t)h^t(n)$$

Design Equations

$$A = \begin{bmatrix} 0 & 1 & & & \\ 0 & 0 & 1 & 0 & \\ \cdot & & & & \\ \cdot & & & & \\ \cdot & & & & \\ & & & 1 & \\ -a(n) & & & & -a(1) \end{bmatrix} \quad \underline{c} = \begin{bmatrix} 1 \\ 0 \\ \\ \\ \\ \\ 0 \end{bmatrix} \quad \underline{k}(t) = \begin{bmatrix} k^t(1) \\ \\ \\ \\ \\ \\ k^t(n) \end{bmatrix}$$

$$\underline{k}(t)\,v(t) = A(Q - Q(t))\underline{c} + \sigma^2\,\underline{h}(1); \; Q(0) = 0$$

$$Q = AQA' + \sigma^2\underline{h}(1)\underline{h}'(1)$$

$$v(t) = r(o) - \underline{c}'\,Q(t)\underline{c} \quad ; \; r(o) = \underline{c}'Q\underline{c} + \sigma^2$$

In these equations, $h^t(n)$ is the response of the system of equations at time $t+n$ to an impulse applied at time t, $R^t(n)$ is the state convariance between $\underline{x}(t)$ and $\underline{x}(t+n)$, and $r^t(n)$ is the output covariance between $y(t)$ and $y(t+n)$. If $r^t(n)$ is to equal $r(n)$, to match the second order properties, then $\underline{k}(t)$ and $v(t)$ must be chosen as above in the design equations.

Kalman Predictor

In the innovations representation, identify $\underline{c}'\underline{x}(t+1/t)$ as the prediction $y(t+1/t)$. Then the Kalman fitter equations become simply a rewriting of the innovations representation:

$$y(t+1/t) = c'\,\underline{x}(t+1/t)$$

$$\underline{x}(t+1/t) = A\underline{x}(t/t-1)+\underline{k}(t)[y(t)-\underline{c}'\underline{x}(t/t-1)]$$

$$y(t) = y(t)$$

Fast Algorithm for $\underline{k}(t)$

The gain $\underline{k}(t)$ may be calculated as outlined under the innovations equations by solving a system of Ricatti equations. An alternative is to note that the gain is related to the time varying impulse response of the innovations representation:

$$h^t(n) = \underline{c}'A^{n-1}\underline{k}(t) \ , \ n>0$$

This means the elements of $\underline{k}(t)$ may be read out as follows:

$$\begin{bmatrix} h^t(1) \\ h^t(2) \\ \cdot \\ \cdot \\ \cdot \\ h^t(n) \end{bmatrix} \begin{matrix} = \\ = \\ \\ \\ \\ = \end{matrix} \begin{bmatrix} k^t(1) \\ k^t(2) \\ \\ \\ \\ k^t(n) \end{bmatrix} = \underline{k}(t)$$

As the correlation matrix for $y(t)$ is related to the time varying impulse responses, we may write

$$R = H\ V\ H^T$$

where the $t \times t$ matrices R,V, and H are defined as follows:

$$R = \{r_{|i-j|}\}$$

$$V = \text{diag}\ (v(o),\ldots v(t))$$

$$H = \begin{bmatrix} h^o(0) & & & \\ h^o(1) & h^1(0) & & \\ & h^1(1) & & \\ h^o(t) & h^1(t-1) & & h^t(0) \end{bmatrix}$$

This means the correlation matrix may be factored with the fast impulse response algorithm of LeRoux-Gueguen [6] to obtain H, and n-dimensional columns of H may be picked off to obtain $\underline{k}(t)$.

Scaling and Rounding in the Kalman Predictor

Let ϵ denote quantization step size and $|\epsilon 2^{m-1}|$ the bound on the maximum magnitude that may be represented in an m-bit, signed binary representation. The problem in a finite-word length realization of the Kalman predictor is to scale variables so that the probability of overflow is small.

Scaling

A filter is said to be l_2 scaled if

$$\sigma^2 \sum_{n=0}^{\infty} f_k(n) = |\epsilon\ 2^{m-1}/\delta|^2$$

where δ is a parameter that may be increased to force the norm of f_k to be small, and thereby decrease the probability of overflow. The parameter σ^2 is the variance of the noisy excitation of the filter.

In the Kalman predictor equation we may scale the state vector $\underline{x}(t/t-1)$ by a diagonal scaling matrix S:

$$S = \text{diag}\ (S(1),\ \ldots,\ S(n))$$

The scaled Kalman predictor equations are then

$$\underline{x}(t+1/t) = S^{-1}AS\ \underline{x}(t/t-1) + S^{-1}\underline{k}(t)\ u(t)$$

$$y(t/t-1) = \underline{c}'\ S\ \underline{x}(t/t-1)$$

The l_2 scaling rules for the states of the Kalman predictor are

$$S^{-1}[Q(t)]_{kk}S^{-T} = |\epsilon\ 2^{m-1}/\delta|^2$$

The state variance $Q(t)$ may be computed recursively as outlined previously, or nonrecursively as

$$Q(t) = v(t) \sum_{n=1}^{t} A^{n-1}\ \underline{k}(t-n)\ (A^{n-1}\ \underline{k}(t-n))'$$

The diagonal terms of $Q(t)$ converge monotonically to the upper limit $Q = AQA' + \sigma^2\ \underline{h}(1)\underline{h}'(1)$. A practical procedure is to replace $\bar{Q}(t)$ by Q to obtain the scaling rule

$$S(k) = [Q]_{kk}^{1/2}\ \delta/2^{m-1} \qquad (k=1,2,\ldots n)$$

This result for stationary state space filters is due to Mullis and Roberts [5].

The algorithm for the gain need not be scaled because it is fixed point: all internal variables are bounded by unity in magnitude.

Rounding

The updating of the state vector in the Kalman predictor requires one multiply for the first (n-1) elements and (n+1) multiples for the n^{th} element. By associating a sequence of i.i.d., variance $\epsilon^2/12$, random variables with each fixed point multiply, we generate a mean zero, variance N random vector $\underline{n}(t)$ each time we update:

$$\underline{x}(t+1/t) = S^{-1}AS\ \underline{x}(t/t-1)+S^{-1}\underline{k}(t)u(t)+\underline{n}(t)$$

$$E\ \underline{n}(t)n'(t+k) = N\delta(k)$$

$$N = \frac{\epsilon^2}{12}\begin{bmatrix}1 & & \\ & 1 & \\ & & \end{bmatrix}_{(n+1)}$$

The state variance is now

$$Q(t+1) = (S^{-1}AS)Q(t)(S^{-1}AS)'+v(t)\underline{k}(t)\underline{k}'(t)+N$$

$$Q=(S^{1}AS)Q(S^{-1}AS)'+\sigma^2\underline{h}(1)\underline{h}'(1) + N$$

Numerical Experiments

Refer to the abstract for a summary of how all experiments were conducted. What follows is a brief annotation of Figures 1 through 4.

Experiments were conducted by generating realizations of a stationary time series from the model

$$H(z) = \frac{1-1.75z^{-1} + 0.8z^{-2}}{1-1.5z^{-1} + 1.21z^{-2} - 0.4550z^{-3}}$$

Figure 1 illustrates $\underline{k}(t)$, a 3x1 vector, computed with floating point arithmetic (curves 1-3) and with fixed point arithmetic (curves 4-6). Figure 2 illustrates the innovation variance $v(t)$ computed in floating point (curve 1) and in fixed point. These two curves illustrate that the data-independent fast Kalman gain calculation may be practically computed in fixed point using 16-bits.

Figure 3 illustrates predictions $y(t/t-1)$ in the floating point realization (curve 1) and in the scaled, fixed point realization of the Kalman predictor.

Figure 4 illustrates the corresponding innovations sequences. The floating point and fixed point predictors were run over the same realization of the time series.

References

[1] S. Sigurdsson, "Fast Kalman Filtering for ARMA Processes: Fixed Point Implementation," M.S. Thesis, Colorado State University, Ft. Collins, CO 80523 (June 1982).

[2] C. Gueguen and L.L. Scharf, "Exact Maximum Likelihood Identification of ARMA Models: A Signal Processing Perspective," Proc. EUSIPCO, Lausanne, Switzerland (September 1980).

[3] J.P. Dugre, L.L. Scharf, and C.J. Gueguen, "Exact Likelihood for Stationary Vector Autoregressive Moving Average Processes," Workshop on Fast Algorithms in Linear Systems, Aussois, France (September 1981).

[4] L.B. Jackson, "On the Interaction of Roundoff Noise and Dynamic Range in Digital Filters," BSTJ, 49 (1970).

[5] C.T. Mullis and R.A. Roberts, "Synthesis of Minimum Roundoff Noise Fixed Point Digital Filters," IEEE Trans. Circuits and Systems, CAS-23 (September 1976).

[6] J. LeRoux and C.J. Gueguen, "A Fixed Point Computation of Partial Correlation Coefficients," IEEE Trans. Acoust., Speech, and Signal Proc., 25, pp. 257-259 (June 1977).

FIGURES

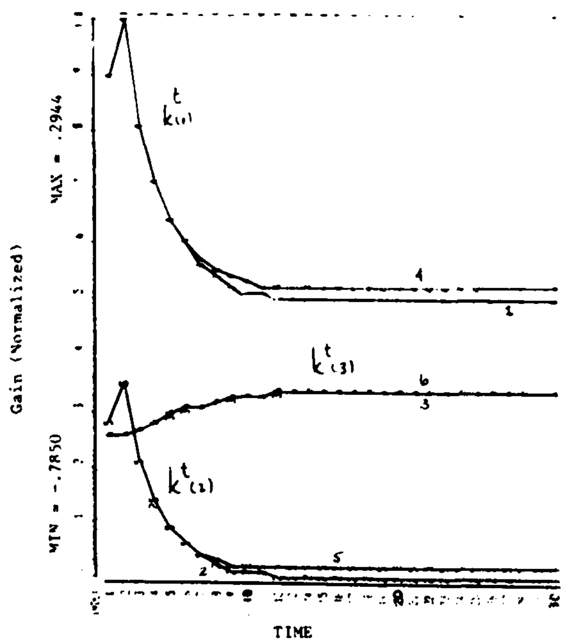


Figure 1. The Kalman predictor gain vector $\underline{k}(t)$ calculated using floating point arithmetic (curves 1-3) and fixed point arithmetic (curves 4-6).

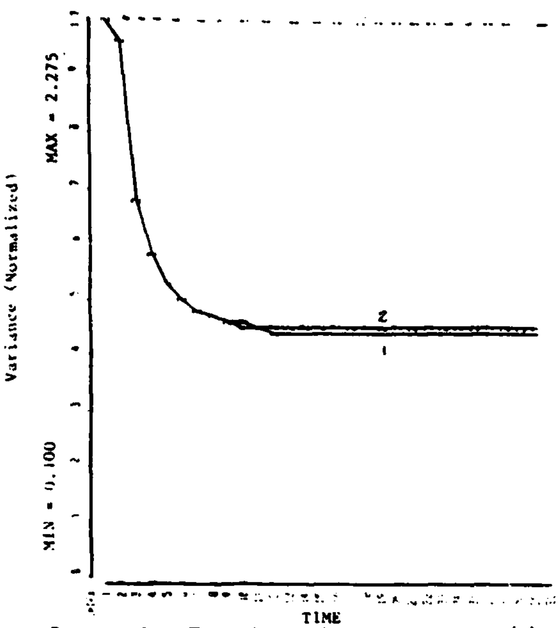


Figure 2. The innovations variance v(t) calculated using floating point arithmetic (curve 1) and fixed point arithmetic (curve 2).

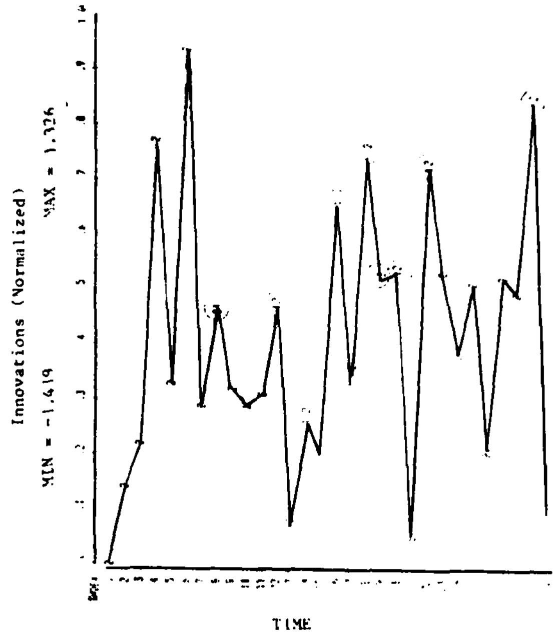


Figure 4. The innovations sequence u(t) calculated using floating point arithmetic (curve 1) and fixed point arithmetic (curve 2).

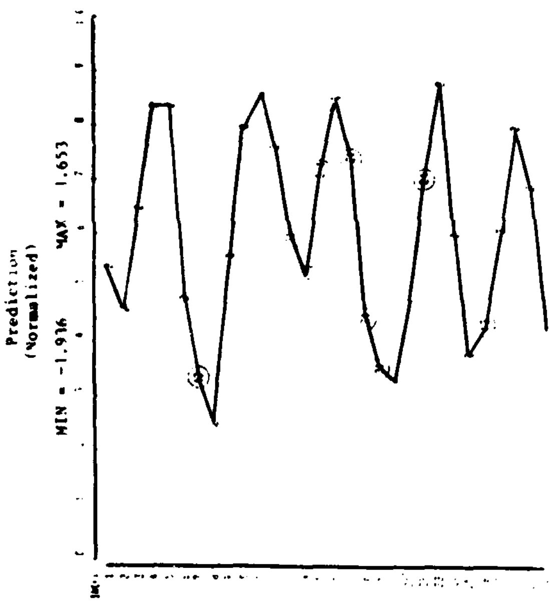


Figure 3. The predicted output y(t/t-1) calculated using floating point arithmetic (curve 1) and fixed point arithmetic (curve 2).

LME

1-84